

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR LETTERS PATENT

**User And Content Aware Object-based Data Stream  
Transmission Methods And Arrangements**

Inventors:  
Huai-Rong Shao  
&  
Ya-Qin Zhang

ATTORNEY'S DOCKET NO. MS1-438US

## **TECHNICAL FIELD**

This invention relates generally to data communications and, more particularly, to prioritization methods and arrangements for selectively transmitting object-based data based on data content in response to dynamically changing network conditions and/or client preferences.

## **BACKGROUND**

Data communication networks are typically divided into two different types of networks based on the techniques used to transmit information between a sending node and a receiving node. The first type or category is a circuit-switched network, wherein a dedicated circuit is established between the sending node and the receiving node prior to the transmission of the information. Information is then transmitted over the dedicated circuit in a continual stream of data. One example of a circuit-switched network is a public-switched telephone network (PSTN). Circuit-switched networks tend to provide a high Quality of Service (QoS) to users since very little, if any, information content is lost during transmission. One of the major drawbacks to circuit-switched networks, however, is that only a limited number of communication sessions or calls can be supported simultaneously by the network. Moreover, in many cases, the communication session fails to fully utilize the available bandwidth of the existing dedicated circuit.

The second type or category is a packet-switched network. In a packet switched network, information is transmitted from the sending node to the receiving node using a plurality of discrete data packets. Each data packet is addressed to the receiving node and selectively forwarded between various nodes

1 within the packet switched network until it reaches the receiving node. At the  
2 receiving node, the received data packets are recombined to provide the original  
3 information. In this manner, the data packets can travel through the network using  
4 a variety of available circuits rather than requiring dedicated circuits. Selective  
5 routing techniques are employed to increase bandwidth utilization within the  
6 packet-switched network. This allows for support of simultaneous communication  
7 sessions, each of which will have an expected QoS. In certain cases, however, the  
8 QoS provided during a communication session may fall below the expected QoS,  
9 for example, due to lost or untimely receipt of data packets. One example of a  
10 packet-switched network is the Internet.

11 The current Internet essentially treats all data packets as having the same  
12 priority during transmission. Therefore, when the Internet is congested, some of  
13 the data packets may be dropped or otherwise lost during transmission. Typically,  
14 such missing data packets do not significantly degrade the overall QoS of the  
15 Internet, since many packets can simply be resent in a timely manner. However,  
16 in time-critical data stream transmissions, such as, multimedia (e.g., video, audio,  
17 etc.) stream transmissions, missing data packets can significantly degrade the  
18 performance of a multimedia application.

19 Consequently, there is a need for improved techniques that significantly  
20 increase the QoS that packet-switched networks, for example, the Internet can  
21 provide to time-critical data streaming applications.

## SUMMARY

Various methods and arrangements are provided to significantly increase the Quality of Service (QoS) provided by packet-switched networks during the transmission of time-critical data streams, and allow for enhanced interactivity for users accessing such time-critical transmissions. For example, prioritization methods and arrangements are provided for selectively transmitting object-based data based on data content in response to dynamically changing network conditions and/or client preferences.

In accordance with certain aspects of the present invention, a method is provided for use in transmitting media information. The method includes receiving a data bitstream that includes object-based media information, associating portions of the object-based media information with a plurality of different transmission priority levels, and selectively transmitting the portions of the object-based media information over a network that is configured to provide differential services based at least on the plurality of different transmission priority levels.

The method may also include receiving at least one down-stream preference with regard to the object-based media information, and selectively transmitting at least one of the portions of the object-based media information over the network based on the down-stream preference.

In certain implementations, the data bitstream includes object-based media information for a single object, such as, a video object or an audio object. The prioritized portions of the object-based media information can be placed in a plurality of data packets, wherein each data packet is associated with a specific

1 transmission priority. The data within a data packet can include non-contiguous  
2 portions of the data bitstream.

3 In certain other implementations, the object-based media information is  
4 encoded using MPEG-4 or other like techniques. Here, media object may be  
5 represented by Intra (I) coded frame layers, Predicted (P) frame layers, Bi-  
6 directionally (B) predicted frame layers, Intra (I) coded frame enhancement layers,  
7 Predicted (P) frame enhancement layers, and Bi-directionally (B) predicted frame  
8 enhancement layers. These frame layers may be associated with specific  
9 transmission priority levels.

10 The object-based media information may further include a plurality of  
11 different types of video object information, such as, e.g., control information,  
12 shape information, motion information and texture information. The transmission  
13 priority level for may also be based at least in part on the type of video object  
14 information.

15 The above stated needs and others are also met by an arrangement that  
16 includes a server device, at least one client device and at least one communication  
17 network. The server device is configured to provide a data bitstream that includes  
18 object-based media information having portions of the object-based media  
19 information associated with a plurality of different transmission priority levels.  
20 The communication network, which is operatively coupled between the server  
21 device and the client device, is configured to provide selective differential services  
22 based at least on the plurality of different transmission priority levels of the  
23 portions of the object-based media information.

24 A system is also provided, in accordance with certain further aspects of the  
25 present invention. In this system, at least one client device is configured to receive

1 prioritized video object-based data packets and output control requests relating to  
2 a video object and at least one server device configured to output prioritized  
3 object-based data packets representing the video object. The prioritized object-  
4 based data packets are prioritized based at least on part on the type of data as  
5 selected from a group comprising control data, shape data, motion data, and  
6 texture data. The system further includes at least one video transmission agent  
7 (VTA), which is coupled to receive the prioritized object-based data packets from  
8 the server device and the control requests from the client device, and to selectively  
9 output at least a portion of the received prioritized object-based data packets to the  
10 client device based in response to the control requests. In certain implementations,  
11 the VTA is implemented within a network that is further configured to provide  
12 differential services to the prioritized object-based data packets, such that  
13 prioritized object-based data packets having lower priority levels are selectively  
14 dropped should the network become congested.

15 In accordance with still further aspects of the present invention, a  
16 computer-readable medium is also provided having a data structure, wherein a first  
17 field contains identifying data associated with a portion of a data bitstream that  
18 represents a video object, and at least one second field that is derived from the first  
19 field and includes data representing object-based video information for the video  
20 object that has been classified as having a specific transmission priority level  
21 based on at least one type of object-based video information selected from a group  
22 comprising control information, shape information, motion information, and  
23 texture information.

## **BRIEF DESCRIPTION OF THE DRAWINGS**

A more complete understanding of the various methods and arrangements of the present invention may be had by reference to the following detailed description when taken in conjunction with the accompanying drawings wherein:

Fig. 1 is a block diagram depicting an exemplary communications system having a network that is configured to provide data bitstream communication services between at least one server device and a plurality of client devices, in accordance with certain aspects of the present invention.

Fig. 2 is a block diagram depicting an exemplary computer that can be employed as a server device and/or a client device in the communications system of Fig. 1.

Fig. 3 is a block diagram depicting a user-aware object-based video multicasting system architecture that can be employed in the communications system of Fig. 1.

Fig. 4 is a block diagram that illustratively depicts an exemplary method/arrangement for prioritizing and packetizing a data bitstream for transmission over the communications system of Fig. 1.

Fig. 5 is a block diagram that illustratively depicts an exemplary format of a prioritized data packet that can be used in the method/arrangement of Fig. 4.

Fig. 6 is a block diagram illustratively depicting certain packetized data paths and associated control paths in the multicasting system architecture of Fig. 3.

Fig. 7 includes two exemplary video frame images for comparison purposes following a simulated loss of 4.1 % of the data packets during transmission, wherein the frame image on the left is noticeably degraded due to the lost data, while the frame image on the right is significantly less degraded due to lost data as

1 a result of the method/arrangement for prioritizing and packetizing a data  
2 bitstream.

3 Fig. 8 is a line graph depicting a measured video quality (PSNR) for the  
4 two images in Fig. 7 as compared to an original image.

5 Fig. 9 includes two exemplary video frame images for comparison purposes  
6 following a simulated loss of 24.3 % of the data packets during transmission,  
7 wherein the frame image on the left is noticeably degraded due to the lost data,  
8 while the frame image on the right is significantly less degraded due to lost data as  
9 a result of the method/arrangement for prioritizing and packetizing a data  
10 bitstream.

11 Fig. 10 is a line graph depicting a measured video quality (PSNR) for the  
12 two images in Fig. 9 as compared to an original image.

13 Fig. 11 is a line graph depicting a measured video quality (PSNR)  
14 associated with a plurality of different data packet loss rates for the two images in  
15 Figs 7 and 9, as compared to an original image.

## 16 17 **DETAILED DESCRIPTION**

18 Fig. 1 is a block diagram depicting an exemplary communications system  
19 10 having a packet-switched network 11 that is configured to provide packet-  
20 switched communication services between at least one server device 12 and a  
21 plurality of client devices 13(a) ... 13(n), in accordance with certain aspects of the  
22 present invention. Although network 11 is referred to as a "packet-switched"  
23 network, portions of network 11 may also include circuit-switched links. For  
24 example, client device 13a may be operatively coupled to network 11 through a  
25 dedicated circuit-switched telephone line. In certain implementations, for



1 example, at least a portion of network 11 can provide differential service  
2 capabilities.

3 As depicted, server 12 is configured to generate or otherwise provide object  
4 data 14 to an encoder 15. The object data can include "media data", such as, video  
5 data, image data, audio data, text data, computer instructions, associated control  
6 data, etc. The media data can include real world information (e.g., still/moving  
7 image from a digital camera) and/or synthetic information (e.g., virtual images).  
8 Object data 14 can represent a two-dimensional or three-dimensional object.

9 Encoder 15 is configured to encode the object data into one or more data  
10 bitstreams. Encoder 15, in certain implementations, is a Moving Picture Experts  
11 Group (MPEG) based encoder. MPEG is the name of a family of standards used  
12 for coding audio-visual information (e.g., movies, video, music) in a digital  
13 compressed format. Preferably, encoder 15 includes the requisite hardware and/or  
14 software to meet the standards of the MPEG-4 or later standards, or other similar  
15 standards that encode media objects into data bitstreams.

16 One advantage of MPEG compared to other video and audio coding  
17 formats is that MPEG files are much smaller for the same quality. This is because  
18 MPEG uses very sophisticated compression techniques to code frames, or as is the  
19 case in MPEG-4 to code objects as separate frame layers.

20 There are three types of coded frames. The first type is an "I" or intra  
21 frame, which is a frame coded as a still image without using any past history. The  
22 second type is a "P" or Predicted frame, which is predicted from the most recent I  
23 frame or P frame. Each macroblock of data within in a P frame can either come  
24 with a vector and difference differential cosine transform (DCT) coefficients for a  
25 close match in the last I or P, or it can be "intra" coded (e.g., as in the I frames).

1 The third type is a "B" or bi-directional frame, which is predicted from the closest  
2 two I frames or P frames, e.g., one in the past and one in the future. There are  
3 typically 12 frames from I frame to I frame (e.g., a sequence of frames may be  
4 ...IBBPBBPBBPBBIBBPBBPB...). Additionally, enhancement I, P, or B frame  
5 layers may be provided to add additional refinement/detail to the image. These  
6 and other features of the MPEG standard are well known.

7 MPEG-4 provides the capability to further define a scene as having a one or  
8 more media objects. For video object, each of these media objects is encoded into  
9 a corresponding elementary data bitstream using I, P, B, and enhancement frame  
10 layers. In this manner, MPEG-4 (and other similarly arranged standards) can be  
11 dynamically scaled up or down, as required, by selectively transmitting elementary  
12 bitstreams to provide the necessary multimedia information to a client  
13 device/application.

14 Server device 12 further includes a prioritizer 16 that is configured to  
15 receive at least one data bitstream from encoder 15. Prioritizer 16 examines the  
16 data bitstream and reconfigures the data bitstream in a plurality of data packets  
17 based on data content. For example, in certain implementations, prioritizer 16  
18 considers the frame type (e.g., I, P, B, etc.) when reconfiguring the data bitstream  
19 into a plurality of data packets. In certain other exemplary implementations,  
20 prioritizer 16 considers the content of the data (e.g., control, shape, motion,  
21 texture, etc.) when reconfiguring the data bitstream into a plurality of data packets.  
22 In still other implementations, prioritizer 16 considers both the frame type and the  
23 content of the data when reconfiguring the data bitstream into a plurality of data  
24 packets. In any of these cases, prioritizer 16 may also consider feedback/control  
25

1 inputs received from other devices/programs, for example, a client device 13  
2 and/or a network 11 resource.

3 In this manner, prioritizer 16 can be arranged to reconfigure the data  
4 bitstream into a plurality of "prioritized" data packets. Prioritizer 16 can  
5 dynamically adjust the prioritization scheme based on inputs received from  
6 resources within network 11, and/or from inputs received from a client 13. These  
7 and other features will be described in future detail below.

8 By way of further example, the information in the data bitstream of object-  
9 based video coding, such as, MPEG-4, can be divided into the following types of  
10 information:

- 11 1. Control information, (e.g., Video Object Head, Video Object Layer  
12 Head and Video Object Plane head);
- 13 2. Shape information of an Intra (I) coded frame base layer
- 14 3. Texture DC information of an Intra (I) coded frame base layer
- 15 4. Texture AC information of an Intra (I) coded frame base layer
- 16 5. Shape information of a Predicted (P) frame base layer
- 17 6. Motion information of a Predicted (P) frame base layer
- 18 7. Texture information of a Predicted (P) frame base layer
- 19 8. Shape information of a Bi-directionally (B) predicted frame base  
20 layer
- 21 9. Motion information of a Bi-directionally (B) predicted frame base  
22 layer
- 23 10. Texture information of a Bi-directionally (B) predicted frame base  
24 layer
- 25 11. Intra (I) coded frame enhancement layer

12. Predicted (P) frame enhancement layer

13. Bi-directionally (B) predicted frame enhancement layer

The preceding types of information can be selectively classified as having different transmission priority levels. Thus, for example, without considering the enhancement layers, the base layers of compressed information may be classified as follows:

CLASS 0: Control information.

CLASS 1: Shape information of the Intra (I) coded frame base layer and texture DC information of the Intra (I) coded frame base layer.

CLASS 2: Texture AC information of the Intra (I) coded frame base layer.

CLASS 3: Shape information of the Predicted (P) frame base layer and motion information of the Predicted (P) frame base layer.

CLASS 4: Texture information of the Predicted (P) frame base layer.

CLASS 5: Shape information of the Bi-directionally (B) predicted frame base layer, motion information of the Bi-directionally (B) predicted frame base layer and texture information of the Bi-directionally (B) predicted frame base layer.

Prioritizer 16, having examined the data bitstream and packetized or otherwise rearranged the data according to different priority levels, then sends the resulting data packets to network 11. If network 11 cannot satisfy the rate requirement associated with the resulting data packets, then data packets having the lowest transmission priority level will be discarded, as needed.

To maintain compliance with MPEG-4 syntax, an index table for each video object is generated without defining a new syntax of the video object bitstream. The index table includes several items, such as index number, information category, priority level, starting position (relative) and length, to index different information in the compressed bitstream within the data packets. The index table is a virtual table that is used only as a reference for extraction of different parts of the information and doesn't constitute part of the bitstream. By way of example, the data structure of the index item may be implemented as follows:

```
typedef struct
{
    char name[30];           // information category
    int priority;            // class level
    int startpos;            // starting position of this part information within the
bitstream
    int length;              // length of this part information
    bool VOPEndFlag;         // label of the end of a frame of the video object
} IndexItem;
```

Each of the client devices 13(a)-13(n) includes a decoder 17 that is configured to receive data from network 11. In certain implementations, for example, decoder 17 is configured to receive at least a portion of the data packets from prioritizer 16, regenerate at least a portion of the encoded data bitstream, decode the resulting data bitstream, and output all or part of object data 14.

In certain other implementations, one or more resources within network 11 may be configured to regenerate at least a portion of the encoded data bitstream

1 using at least a portion of the data packets output by prioritizer 16. For example, a  
2 server, gateway, router, or other like device may be provided within network 11 to  
3 perform this task. In still other implementations, prioritizer 16 may alternatively  
4 be included at least partially within various network 11 resource(s).

5 Reference is now made to Fig. 2, which is a block diagram depicting an  
6 exemplary computer 20 that can be employed within server device 12 and/or client  
7 devices 13(a)-13(n), for example, to provide or otherwise support data bitstream  
8 communications. Those skilled in the art will clearly recognize that different  
9 computing and communication resources and configurations can be operatively  
10 configured to form computer 20 or otherwise perform the functions of network 11,  
11 server device 12, and client devices 13(a)-13(n). For example, a client device 13  
12 may include a cellular telephone, a pager, or other wireless device that is  
13 operatively coupled to network 11.

14 As shown in this example, computer 20 includes one or more processors or  
15 processing units 21, a system memory 22, and a bus 23 that couples various  
16 system components including the system memory 22 to processors 21. Bus 23  
17 represents one or more of any of several types of bus structures, including a  
18 memory bus or memory controller, a peripheral bus, an accelerated graphics port,  
19 and a processor or local bus using any of a variety of bus architectures.

20 The system memory includes read only memory (ROM) 24 and random  
21 access memory (RAM) 25. A basic input/output system (BIOS) 26, containing the  
22 basic routines that help to transfer information between elements within computer  
23 20, such as during start-up, is stored in ROM 24.

24 Computer 20 further includes a hard disk drive 27 for reading from and  
25 writing to a hard disk, not shown, a magnetic disk drive 28 for reading from and

1 writing to a removable magnetic disk 29, and an optical disk drive 30 for reading  
2 from or writing to a removable optical disk 31 such as a CD ROM, DVD ROM or  
3 other optical media. The hard disk drive 27, magnetic disk drive 28 and optical  
4 disk drive 30 are each connected to bus 23 by applicable interfaces 32, 33 and 34,  
5 respectively.

6 The drives and their associated computer-readable media provide  
7 nonvolatile storage of computer readable instructions, data structures, program  
8 modules and other data for computer 20. Although the exemplary environment  
9 described herein employs a hard disk, a removable magnetic disk 29 and a  
10 removable optical disk 31, it should be appreciated by those skilled in the art that  
11 other types of computer readable media which can store data that is accessible by a  
12 computer, such as magnetic cassettes, flash memory cards, digital video disks,  
13 random access memories (RAMs) read only memories (ROM), and the like, may  
14 also be used in the exemplary operating environment.

15 A number of program modules may be stored on the hard disk, magnetic  
16 disk 29, optical disk 31, ROM 24, or RAM 25, including an operating system 35,  
17 one or more application programs 36, other program modules 37, and program  
18 data 38. A user may enter commands and information into computer 20 through  
19 input devices such as keyboard 40 and pointing device 42. Other input devices  
20 (not shown) may include a microphone, joystick, game pad, satellite dish, scanner,  
21 or the like. These and other input devices are connected to the processing unit 21  
22 through an interface 46 that is coupled to bus 23.

23 A monitor 47 or other type of display device is also connected to bus 23 via  
24 an interface, such as a video adapter 48. In addition to the monitor, personal  
25

1 computers typically include other peripheral output devices (not shown) such as  
2 speakers and printers.

3 Computer 20 can operate in a networked environment using logical  
4 connections to one or more remote computers, such as a remote computer 50.  
5 Remote computer 50 may be another personal computer, a server, a router, a  
6 network PC, a peer device or other common network node, and typically includes  
7 many or all of the elements described above relative to computer 20. The logical  
8 connections depicted in Fig. 2 include a local area network (LAN) 51 and a wide  
9 area network (WAN) 52. Such networking environments are commonplace in  
10 offices, enterprise-wide computer networks, intranets, and the Internet.

11 When used in a LAN networking environment, computer 20 is connected to  
12 the local network 51 through a network interface or adapter 156. When used in a  
13 WAN networking environment, computer 20 typically includes a modem 54 or  
14 other means for establishing communications over the wide area network 52, such  
15 as the Internet. Modem 54, which may be internal or external, is connected to bus  
16 23 via interface 46. In a networked environment, program modules depicted  
17 relative to the personal computer 20, or portions thereof, may be stored in the  
18 remote memory storage device. It will be appreciated that the network  
19 connections shown are exemplary and other means of establishing a  
20 communications link between the computers may be used.

21 Future generations of the Internet Protocol will provide differential service,  
22 whereby data packets will be treated as having a specified priority. Thus, for  
23 example, when the Internet or a portion thereof is congested, data packets with  
24 lower priorities may get dropped/delayed to better insure that packets with higher  
25 priorities get delivered. This allows for different QoS's to be provided based on



1 the importance associated with the user and/or information content. The QoS  
2 class of each data packet can be specified, for example, using a type of service  
3 (TOS) byte within the Ipv4 header or a traffic class byte within the Ipv6 header  
4 (IPH) of the data packet.

5         Prioritization of object-oriented video information may be advantageously  
6 implemented using this type of differential service capability to provide a scalable  
7 video differentiated transmission scheme in which media objects that are identified  
8 in a video frame, or portions thereof, are associated with various differential  
9 priorities during transmission. Thus, for example, different kinds of information,  
10 such as, shape, motion and texture information, within the data bitstream are  
11 associated with specific transmission priority levels. In certain implementations,  
12 the transmission priority levels associated with the information will significantly  
13 match the level of importance that the information inherently has within the  
14 decoding process in decoder 17. Thus, for example, shape information as  
15 described in the earlier example may be more important than motion or texture  
16 information.

17         The resulting differentially prioritized data stream has many benefits. For  
18 example, the resulting prioritized data stream can be used to better control data  
19 flow at server 12 or within network 11, meet QoS requirements, enhance user  
20 interaction capabilities, and/or better support multicasting services to a plurality of  
21 receiving nodes 13(a)-13(n).

22         With this in mind, reference is now made to Fig. 3, which is a block  
23 diagram depicting a user-aware object-based video multicasting system  
24 architecture that can be implemented in communications system 10.  
25

1 In Fig. 3, server device 12 is operatively coupled to a first network node  
2 82a. Here, network nodes 82a-d can be any type of general purpose and/or special  
3 purpose computing/communication device, such as, a server, a gateway, a router, a  
4 computer, etc. Client device 13a is operatively coupled to receive data packets  
5 from network node 82a. The output of network node 82a is provided to a first  
6 video transmission agent (VTA) 84a. VTA 84a is configured selectively filter out  
7 or otherwise drop data packets according to their respective transmission priority  
8 levels based on network and/or other recognized needs (e.g., downstream  
9 feedback/control).

10 The output from VTA 84a is provided to a second network node 82b, which  
11 is operatively coupled to provide data packets to client devices 13b, 13c and third  
12 network node 82c. Network node 82c is operatively coupled to provide data  
13 packets to client device 13d and a second VTA 84b. VTA 84b is configured to  
14 selectively filter out or otherwise drop data packets according to their respective  
15 transmission priority levels based, again, on network and/or other recognized  
16 needs. The output from VTA 84b is provided to a fourth network node 82d, which  
17 is operatively coupled to provide data packets to client devices 13e and 13f.

18 An optional third VTA 84c is also shown as being included within network  
19 node 82a. VTA 84c may be operatively configured to selectively control the data  
20 packets that are provided to client device 13a.

21 Fig. 4 is a block diagram that illustratively depicts an exemplary method  
22 and/or arrangement for prioritizing and packetizing media object data from an  
23 encoded bitstream. As shown, an encoded data bitstream 100 having data 102 is  
24 provided. Next, the data bitstream is examined to identify portions 104a-e, each of  
25 which is associated with a specific transmission priority level. The portions 104a

1 through 104e are then placed in data packets 106a-c having a corresponding video  
2 packet header (VPH) 110.

3 As shown in this example, portion 104a has a priority level of "P0", portion  
4 104a has a priority level of "P0", portion 104b has a priority level of "P1", portion  
5 104c has a priority level of "P2", portion 104d has a priority level of "P2", and  
6 portion 104e has a priority level of "P0". Thus, data packet 106a includes portions  
7 104a and 104e, each of which has a transmission priority level of P0. Identifying  
8 bits 107 are provided in data packet 106a to identify that portions 104a and 104e  
9 are not contiguous within data bitstream 100.

10 Similarly, data packet 106b includes portion 104b, which has a  
11 transmission priority level of P1, and data packet 106c includes portions 104c and  
12 104d, each of which has a transmission priority level of P2.

13 Data packets 106a-c are then each further configured for transmission over  
14 network 11, for example, by adding a Network header (NPH) 114 to create data  
15 packets 108a-c, respectively. By way of example, NPH may be TCP Header +IP  
16 Header, UDP header +IP Header, IP Header, or other format. . NPH 114 identifies  
17 the transmission priority level.

18 An exemplary data packet 106a is depicted in Fig. 5. As shown, VPH 110  
19 includes an object identifier (OI) 110a, a part number (PN) 110b and a starting  
20 position (SP) 110c for the first portion (or contiguous portions) of data in data  
21 packet 106a (i.e., here, portion 104a). Additionally, a length (LEN) 112a is  
22 provided to establish the length of the first portion (or contiguous portions).  
23 Additional SP 110c and LEN 112b data are provided following identifying bits  
24 107 to further define portion 104e. The identifying information in 106a can be  
25

1 used to reconstruct the data bitstream 100, or remaining portion thereof, after  
2 transmission over network 11.

3 In certain implementations, a video packet 106 is limited in size so as to not  
4 exceed the MTU of network 11. Additionally, a video packet 106 should not be  
5 too small to increase efficiency.

6 Fig. 6 is a block diagram illustratively depicting how a plurality of VTAs  
7 can be used to control the flow of packetized data between server device 12 and a  
8 plurality of clients 13a, 13b, 13c, and 13e with respect to multicasting of video  
9 data. In this example, the various nodes have bi-directional feedback/control  
10 communication capabilities through which client device software is configured to  
11 relay user inputs with respect to selected media objects and/or other related  
12 interactive features. Thus, for example, a user may select a specific video object  
13 within a scene through a graphical user interface. This selection will be sent  
14 upstream to one or more VTAs and possibly to the server device 12 itself, if  
15 needed, to elicit the proper response in the packetized data that is sent  
16 downstream.

17 In this manner, the VTAs are able to respond dynamically to the needs of  
18 the client devices. As mentioned above, the VTAs may also respond to network  
19 related constraints. Thus, both network and user interactivity behaviors are taken  
20 into consideration.

21 The user can interact with a video player application program running on  
22 the client device, for example, and the server device through several ways such as  
23 mouse click, mouse moving, forward, backward, object zoom in, object zoom out,  
24 add objects, delete objects, etc. These interaction behaviors can influence the  
25 amount of network bandwidth allocated to individual object dynamically.

1 Accordingly, the control paths provide a mechanism for negotiating or  
2 otherwise establishing what object data 14 needs to be transmitted and how best to  
3 allocate limited bit rate among multiple video objects according to network state  
4 and users' interaction.

5 By way of further example, let us assume that server device 12 is  
6 multicasting object-based MPEG-4 video to the four clients. Server device 12 will  
7 packetize the bitstream of each requested video object and send data packets with  
8 different transmission priority levels (e.g., different QoS requirements). Should  
9 congestion occur within network 11 during transmission, one or more network  
10 nodes 82 may need to discard data packets having lower priority levels.

11 The VTAs will filter out data packets from upstream according to  
12 downstream requirements (e.g., according to the network link and a particular  
13 clients capabilities). For example, data packets containing texture information for  
14 a particular video object may be dropped by the VTA.

15 This capability is graphically depicted by the varying width of data flow  
16 arrows 120, 122, 124, 126, 128, 130, and 132; here, wider data arrows represent  
17 more data packets being transmitted. Thus, for example, VTA 84c (and possibly  
18 other network resources) has dropped some of the data packets received from  
19 server device 12 prior to providing the remaining data packets to VTA 84c, since  
20 arrow 120 is wider than arrow 124.

21 With this in mind, a new scalable video transmission technique is provided  
22 in which the user's interaction and at least portions of the video information  
23 content itself can be taken into consideration. This novel technique allows for  
24 efficient object-based video transmission over IP based networks or other network  
25 architectures that provide differentiated services.

1 The techniques include smart packetization processes that distinguish  
2 between different types of data in a bitstream, object-based bit rate control and  
3 adaptation processes corresponding to users' interactions and selections, and  
4 selective transmission processes that drop data packets as needed to take  
5 advantage of differentiated classes. The packetization scheme can take advantage  
6 of object boundary, MB (macro block)/MV (motion vector) structure, shape  
7 information, and the nature of differential services architecture. The bit rate of a  
8 video stream can be adjusted to network state by discarding low priority packets.

9 The VTAs represent a novel object-based video streaming mechanism  
10 that has capability of providing dynamic interaction between users and video  
11 contents. A VTA can dynamically adapt the bit rate of each video object to the  
12 user's interactivity behaviors.

13 As demonstrated by the foregoing detailed description and drawings, the various  
14 methods and arrangements can be implemented to provide error control and  
15 scalability for compressed video data over the Internet. During transmission, the  
16 data within a compressed video bitstream is packetized and the packets are sent to  
17 the receiver over the Internet. Although some of the data packets may be lost  
18 during transmission, their loss will not necessarily lead to failure in the decoding  
19 process since different types of data within the data bitstream, such as, e.g., shape,  
20 motion and texture information have been essentially re-clustered and assigned to  
21 different transmission priority levels/classes. Consequently, higher priority classes  
22 of data packets will have lower packet loss rate than relatively lower priority  
23 classes of data and obtain increase error control protection.

24 This prioritization technique has the additional benefit of allowing changes to be  
25 made to the transmission rate of the video just by selectively discarding lower

1 priorities packets at server 12 and/or at intermediate network nodes 82. Moreover,  
2 the methods and arrangements are particularly useful in supporting multicasting  
3 applications/services.

4 Multicasting is regarded as an effective communication support for multi-party  
5 multimedia applications, such as, distance learning and video broadcasting.

6 However, due to the heterogeneity of the Internet a single sender transmission rate  
7 cannot satisfy the different bandwidth requirements of different clients. Typically  
8 the sender rate is adapted to meet the requirements of the worst positioned client,  
9 thereby reducing the quality of the video data for other clients. While such  
10 limitations may be partially overcome using conventional layered transmission  
11 mechanisms, wherein each layer (elementary bitstream) is sent over a separate  
12 network session, it is complicated for the network to maintain multiple sessions  
13 for each video object and the synchronization control between different layers is  
14 difficult to achieve. In addition, the transmission rate cannot be adjusted to a  
15 granularity that is smaller than the difference between layers. The methods and  
16 arrangements described herein provide a novel heterogeneous multicasting  
17 mechanism that solves these and other problems within layered and worst-case  
18 multicasting approaches. Through this prioritized single layering approach  
19 different clients can receive different quality levels of multimedia data bitstreams.

20 With respect to video, for example, after encoding/compressing, the  
21 compressed video data are placed in a data bitstream according to temporal and  
22 spatial position of its content, i.e., frame by frame, macroblock by macroblock,  
23 and block by block, etc.

24 Traditionally, different types of information such as shape, motion and  
25 texture information would have been interleaved together at this stage, despite the

fact that each has an inherent and different level of importance during the decoding process. For example, shape and motion information is usually more important than texture within a P frame. Consequently, if the shape and motion information is lost during transmission, decoder 17 will be unable to output the uncompressed P video frame even though it received texture information. However, shape and motion information may be enough for decoder 17 to generate a coarse video frame that will have a lower quality than the original frame due to the loss of texture information.

Moreover, at present, rate control for video is mainly implemented by changing the encoding quantization size, selecting another optimal mode or reducing the frame rate within encoder 15. For many applications 36, such as, for example, video on demand (VOD), video materials are encoded and stored at server 12 in advance. Therefore, the bit rate cannot be subsequently changed to adjust the network resource availability. For those adopting real-time encoding, if the encoded video needs to be sent to multiple heterogeneous receivers simultaneously, the single bit rate may not satisfy all receivers. By employing a smart prioritization/packetization and rate control approach the methods and arrangements can be better configured to control and respond to the communication needs of the clients. For example, the methods and arrangements can take advantage of object boundary, MB (Macro Block)/MV (Motion Vector) structures, shape information, and even future generations of the Internet infrastructure which is expected to provide differentiated services.

Thus, when network resources are unable satisfy the rate requirement(s) of the video bitstream, information with the lower priorities will be selectively discarded by server 12 or intermediate network nodes. In addition, different error



1 control mechanisms may be implemented on different priorities information to  
2 further enhance the overall error resilience capability of the transmission.

3 By way of further example, since the rate control is supported by discarding  
4 some information, let us assume that there are N priority levels  $P_i$  ( $0 \leq i < N$ ),  
5 each level information has the original bit rate  $r_i$  ( $0 \leq i < N$ ), and the original rate  
6 of the whole video stream is R. Therefore,

$$7 \quad R = \sum_{i=0}^{N-1} r_i$$

8  
9 During transmission, if network congestion occurs and/or the receivers  
10 require lower bit rates (e.g., as specified through some sort of user-interactivity  
11 behavior), then the bit rate for the video object needs to be reduced to  $R'$ .

12 First, the suitable  $k$  ( $0 < k < N$ ) needs to be found which satisfy the following  
13 form:

$$14 \quad \sum_{i=0}^{k-1} r_i \leq R' < \sum_{i=0}^k r_i$$

15  
16 Wherein, all information with priorities  $P_j$  ( $k < j < N$ ) will be discarded at the  
17 sender.

18  
19 Secondly, if

$$20 \quad R' < \sum_{i=0}^{k-1} r_i,$$

21  
22  
23 all packets with priorities  $P_k$  will also be discarded at the sender; otherwise,  
24 some fine bit rate adjustment will be implemented. This means some packets with  
25 priority  $P_k$  will be selectively discarded at the sender and/or the intermediate

nodes. For example, if some information within a B frame needs to be dropped, then a selective B frame discarding scheme as proposed by Hemy et al., "MPEG System Streams in Best-Effort Networks", Packet Video'99, New York, can be adopted. In another example, if texture information within a P frame needs to be partly discarded, the approach can randomly discard "A" percentage of P frame texture information, wherein:

$$A = \frac{\sum_{i=0}^k r_i R'}{r_k} \times 100\%$$

As can be seen, the rate control mechanism provides for both the temporal scalability and quality scalability support.

Moreover, as described above, such a scalable transmission scheme can also take advantage of differentiated services by providing the selective filtering capability of a VTA 84. The resulting layer approach allows different users to receive different quality levels of video data bitstreams. VTA 84 and associated (Capacity, Requirement) resource allocation policy may be implemented to provide a more flexible multicasting mechanism.

Here, an exemplary VTA 84 is configured to receive video data bitstreams from server 12 or a previous VTA, filter the received data bitstream by selectively discarding data packets having lower priorities, and sending the resulting filtered data bitstream to a client 13 and/or a next (downstream) VTA 84. VTA 84 may also be configured to receive requests from one or more clients and/or other VTAs, act upon requests or generate a combined request from requests of multiple clients or next VTAs and forward it to previous VTAs, provide bandwidth allocation coordination among multiple data bitstreams, and/or even provide

1 dynamic bandwidth re-allocation according to the semantic information of video  
2 object, such as, for example, when a scene change is identified within the data  
3 bitstream as occurring.

4 By way of further example, VTA 84 can be implemented as a component of  
5 the router or gateway, a server or part of a server attaching to the router, and  
6 configured to watch a known port, intercept requests and adapt the data bitstream  
7 accordingly. Thus, as depicted in Figs 3 and 6, VTAs can be implemented at  
8 various nodes within network 11 to provide for multiple filters along the path from  
9 server 12 to client 13. A resource allocation policy, e.g., (Capacity, Requirement)  
10 policy, can be employed within VTA 84 to support multicasting. For example, in  
11 a multicasting session, every VTA and client is associated with a two element set,  
12 namely "Capacity" and "Requirement". Capacity represents the amount of the  
13 resource available for a video object in the VTA or the end system of the user to  
14 handle the video object. Requirement represents the amount of the resource for a  
15 video object required by the VTA or the user. Concerning the network bandwidth,  
16 the Capacity of a VTA is determined by the link state between the VTA and its  
17 next upstream VTA (or the server 12 if it has no upstream VTA). Let us assume  
18 that the capacity of the sender is the bit rate of the original bitstream, denoted as  
19  $RATE_s$ . However, the Requirement of a VTA is determined by (Capacity,  
20 Requirement)<sub>s</sub> of all its next downstream VTAs and the clients connecting directly  
21 to this VTA. For example, if a VTA, named  $V_i$ , has  $M$  next downstream VTAs and  
22 clients whose two element set are  $(Cap_{ij}, Req_{ij})$  ( $0 \leq j < M$ ), then the  
23 Requirement of  $V_i$  is calculated as follows:

$$24 \quad Req_i = \min \{ \min(Cap_{ij}), \max(Req_{ij}), RATE_s \} (0 \leq j < M).$$

25

1 For a client, Requirement will likely be determined by the user's preference  
2 and interactivity behavior.

3 For a multicasting session, the (Capacity, Requirement)<sub>s</sub> of its VTAs and  
4 clients can be calculated following the direction from bottom to top (i.e., moving  
5 upstream). And at each VTA<sub>i</sub>, the rate of the video object bitstream can be adapted  
6 to Req<sub>i</sub> by using the scalable transmission scheme proposed herein.

7 As can be seen, the VTAs have the functionality of filtering. However,  
8 traditional video filter needs to implement decoding and re-encoding to complete  
9 the filtering. Here, however, the VTAs can implement the filtering by only  
10 discarding some less important packets in a much simpler and faster fashion than  
11 traditional filters.

12 In accordance with certain experimental implementations, a simple MPEG-  
13 4 video streaming system is used to test a scalable multicasting scheme. The test  
14 bed included MPEG-4 video encoding/decoding source code and Ipv6 protocol  
15 stack source codes. A standard Akiyo QCIF video sequence was used in the  
16 experiment. Exemplary results from the experiment are depicted in the  
17 images/graphs in Figs 7-11.

18 As shown in the two examples of Figs 7, 8 and Figs 9, 10 the approaches  
19 taught by the methods and arrangements perform better than traditional  
20 approaches under the same network bandwidth and packet loss conditions.

21 The images in Fig. 7 were produced with a 4.1% data packet loss rate and  
22 represent frame number 81. The image on the left was produced using a  
23 traditional approach (e.g., random loss), and the image on the right was produced  
24 as a result of using exemplary methods and arrangements as taught herein. Fig. 8  
25 shows comparison of peak Signal to noise luminance parameter for every frame

1 between the traditional approach and the present approach with a 4.1% data packet  
2 loss rate, wherein line 200 depicts the original quality of the video sequence, line  
3 202 depicts the quality resulting from this new approach, and line 204 represents  
4 the quality as a result of a traditional approach.

5 The images in Fig. 9 were produced with a 24.3% data packet loss rate and  
6 represent frame number 81. The image on the left was produced using a  
7 traditional approach, and the image on the right was produced as a result of this  
8 new approach. Fig. 10 shows comparison of peak Signal to noise luminance  
9 parameter for every frame between the traditional approach and the new approach  
10 with a 24.3% data packet loss rate, wherein line 206 depicts the original quality of  
11 the video sequence, line 208 depicts the quality resulting from this exemplary  
12 implementation, and line 210 represents the quality as a result of a traditional  
13 approach.

14 As can be concluded from Fig. 11 the new scalable transmission scheme is  
15 feasible to be used in multicasting mechanism because the PSNR of the video does  
16 not decrease very much even though the network bandwidth decreases to half of  
17 the original video rate.

18 Although some preferred embodiments of the various methods and  
19 arrangements of the present invention have been illustrated in the accompanying  
20 Drawings and described in the foregoing Detailed Description, it will be  
21 understood that the invention is not limited to the exemplary embodiments  
22 disclosed, but is capable of numerous rearrangements, modifications and  
23 substitutions without departing from the spirit of the invention as set forth and  
24 defined by the following claims.